

Praktikum "The Bioinformatics Lab"

Week 1: Install a Debian basic system onto a bootable USB stick

Verena Link

16.05.2011

In the first week of the "Bioinformatics Lab" we should install a Debian Linux system on a bootable USB-Stick. In the following report I will describe how to install this basic system.

1 Boot CD Preparation

The first step of the installation is to download a Debian image from the Debian webpage (www.debian.org). It is important to download an image, which is suitable to the computer architecture.

1.1 Computer architectures and the different Debian images

There are many different Debian images on the debain website, which can be downloaded by the user. These different images fit to different processor architectures. The two most common processor architectures on Latops and Desktop PCs are Intel 32 bit processors or AMD and Intel 64 bit processors. For the Intel 32 bit processors the i386 image is the right choice, whereas for the 64 bit processors the amd64 image is the suitable one. In a case of doubt about the processor architecture it is useful to choose the i386 image, because alle 64 bit architectures can emulate 32 bit processors. It is also possible to look in the BIOS, where you can find out which processor is on your mainboard. I installed the Debian on a Laptop with an Intel 32 bit processor and so I decided to download the i386 image.

1.2 Preparing the CD

Nowadays, most of the people go to the Internet via a DSL connection and therefore, it is useful to choose a netinstall image. For people who don't have a fast Internet connection, it is also possible to download a complete Debian image, which contains all necessary packages. Nevertheless, it is always useful to have an Internet connection when installing Debian, because otherwise the installer won't configure the network by default and the user has to configure the network after the installation process. I chose a netinstall Debian stable 6.0 ("Squeeze") image and downloaded it to my computer. In the next step I

burned this image to a CD and set this CD as bootable, which is possible with all burning programs, like Nero Burning Rom on Windows, Burn on Max OS X or cdrecord on Linux. For more information please read the documentation of your program.

2 Installation

After burning my bootable CD, I chose in my BIOS the Boot option "Boot from CD" in the BIOS boot options to boot from CD and therefore, to start the installer. I chose the expert install mode, because this mode gives better control over the installation. It is also possible to use the graphical installation method or the normal console-based installation method. Regardless of which installation method was used, each method will give you a running Debian system at the end.

2.1 Preparing the installer

As first step of the installation process the installer wants to configure the language of the system and the keyboard layout. I decided to install an english system with german keyboard layout. There are two good reasons for an english system. First of all, Laszlo can help me with problems of the system. But the more important point is, that error message in english are more meaningful than german error messages (which are often translated very badly). Furthermore, you get more hits in google by searching for the english error messages. Next the installer configures the system clock.

2.2 User accounts

The next important step in this installation is to create the users. The first user is the root (administration) user. The root user has full access to every file on the system. So it is advisable choose a very powerful password, which protects your system against crackers. Furthermore, you can - but don't have - to create a user. I created a user called link, which is my login name in the Kramer lab. For your own security choose a password other than that of the root account. In section 3.2.1 I will document how to install sudo, so it is possible to deactivate the root user and change between root and user by using the sudo command. Every linux system has a user called root and so it is easier to crack the root account, because you only have to crack the password, (the username is already known). So using sudo to gain root access is safer than to directly use the root account.

2.3 Network Setup

In the next step of the installation the installer wants to configure the network. It is possible to choose between the configuration of the ethernet card or the wifi card. I decided to configure the ethernet card, because the installer often doesn't have drivers to set up the wifi card. If you have configured a DHCP server in your network you have the possibility to use this server and the installer will configure your network device by itself. Otherwise you have to configure

the network manually, which I chose, because I don't have a DHCP server in my network. Especially for the netboot install (which is the one I used for my setup) you have to configure your network, otherwise you can't install the required packages. Users, who use a network install image instead, should also configure their network, because with the installer it is very easy to configure the network. If you configure your network after installing your Debian, you have to edit different config files and to restart your network services. So especially for users, which install a linux system for their first time, it is very advisable to configure the network during the installation.

2.4 Partitioning

Before it is possible to start the installation process, it is necessary to prepare the USB-Stick (or normally the hard disk) for the installation. Therefore, the installation device has to be partitioned. In our case we should have three different partitions. The first one is the /boot partition, which contains all files which are necessary to boot the system such as the boot loader and the kernel. The second partition is used as swap partition, which is used as temporary storage device. Nowadays, RAM is not that expensive and every computer has a lot of it (normally 2 - 4 GB) so it isn't mandatory to create a swap partition. The third partition we have to create is the "root" / partition, which contains all other files, which are not necessary for the boot process (system files, user files, data etc.). It is also possible to create only two partitions, one for swap and the other one for the data. The second partition contains / and /boot, but this causes troubles when encrypting the "root" partition. Therefore it is more useful to create three different partitions. This week we should install a basic Debian system and the advanced challenge is to install the Debian system onto an encrypted partition. In this report, I will describe both methods. Whatever partitioning schema is used, it is important to be careful when partitioning the device. Especially if you install the Debian system on the USB stick, there is still one working hard disk in your computer which contains your currently operating system. Therefore, it is important to be careful and not to partition the hard disk instead of the USB device. I decided to partition my device manually and not with the "guided partitioning", because the manual method gives me more control over the partition layout. My USB stick was found under /dev/sdb, whereas my hard disk is under /dev/sda. This could be different on other systems, depending on how many hard disks are in the computer (for example a system with two hard disks has disk one on /dev/sda, disk two on /dev/sdb and the USB stick on /dev/sdc).

2.4.1 Unencrypted partitioning

For the unencrypted partitioning, which is the easier method, we need three different partitions - 512MB for swap, 512MB for /boot and the remaining 7GB for the "root" partition /. With the following steps, it is possible to create such a partition schema:

- Locate the USB stick in the partition overview (mostly it is on /dev/sdb). Select the 8GB FAT32 partition which are the USB stick's partition and delete it.

- 8GB free space appears in the overview, which can be partitioned now. Add a new, primary partition with size 512MB at the beginning using ext4 filesystem. As mountpoint choose `/boot` and toggle the bootable flag on yes (just scroll to this point and press return).

After this step there are still 7,5GB remaining. Add a new, primary partition at the beginning as your swap partition. For this purpose choose "swap area" in the field "use as:"

- Select the remaining free space and add a new, primary partition at the beginning using the ext4 file system. As mountpoint choose `/`.
- After checking if all partitions were created correctly on the correct device select "finishing" to write the current partitions on your USB stick

2.4.2 Encrypted partitioning

The advanced challenge of this week was to install the linux system onto an encrypted partition. This is a little bit more complicated than just install the system onto an unencrypted partition, but it is much safer. One possibility to install an encrypted system is to make two encrypted partitions, one for swap and the other one for root, but this means it is necessary to type in two passwords during the boot process. Therefore it is much better to make one encrypted partition and to divide this partition with the LVM (Logical Volume Manager). I used the second method, which is described in the following steps:

- To create an encrypted partition it is necessary to choose the manual partitioning.
- As before, the FAT32 partition on the USB stick has to be deleted.
- First, the `/boot` partition is created. It is not possible to create an encrypted partition for `/boot`, because the system can't read the encrypted boot partition during the boot process. This means a system with an encrypted boot partition can't boot. Add the new primary partition at the beginning and choose ext4 as filesystem. As described before, toggle the bootable flag to yes
- Now create a new partition, which consists of the remaining 7.5GB. Create this partition as a "physical volume for encryption".
- Now choose "Configure encrypted volumes" to save the partition table and to configure the encrypted partition. If there is another unencrypted swap partition on the hard disk, the installer will cause problems, because the swap partition is unencrypted. Change the console and deactivate this swap partition with the command

swapoff /dev/<device>

I had this problem during the installation and it was not possible to deactivate the swap partition, because my system always remounted it. Therefore, I deleted the unencrypted swap partition and recreate it after

installing the new system on the USB stick. The installer will ask for a passphrase, which should also be a very powerful one.

- After this step, the Partition disk view appears again. On the top of the partitions there is now an encrypted volume. It contains an ext3 file system. In order to get two different partitions onto this encrypted partition and to use ext4 instead of ext3 it is now necessary to configure the encrypted partition. Therefore the partition has to be changed in a virtual volume by using the LVM.
- Choose the point "configure the Logical Volume Manager" now to configure the partition. The installer asks if the encrypted volume should become formatted and if the changes should be written to disk to configure the LVM, which I answered with yes.
- Now choose "create volume group" and enter a new name for this group and choose the encrypted partition as device for the new volume group. Next create a logical volume in that volume group with a size of 512MB and call it "swap". In the second step create another logical volume, which takes the rest of the free space and call it "root". Choose "Finish" to save the changes on the encrypted partition.
- Back to the partition disks view there are two new entries. The first entry has a size of 7GB. Choose this volume and use it as ext4. Change the mount point to the "root" directory / and save the changes.
- The other partition has a size of 512MB. Now choose this volume change the use as filed to "swap area" and save the changes.
- Last, check if everything is configured correctly and than choose "Finish partitioning and write changes to disk"

2.5 Installation of the basic system

After partitioning the USB stick, the stick is now ready for a debain installation. First of all the installer installs the basic system, which means it installs the basic packages, the locales, configures the keyboard and so on. After that you have to decide which kernel should be installed. In this case it is almost a good choice to choose the kernel the system propose you. Next the installer asks, if it should install all availabe drivers, or just drivers which are useful for the system. It is a better choice to install all availabe drivers, because then you don't have much problems if you plug in other hardware.

2.6 Mirror Selection

With the now configured network it is possible to download all required packages for the installation. For this purpose, you have to choose a mirror. It is very sensible to choose a mirror, which is located close to you, so the data transfer will be faster.

2.7 Installation of additional packages

After selecting a mirror, the installer shows you a lot of additional packages, which can be installed. Since we want to install a linux server and not a workstation I decided just to install the "Standard-System" and the "Laptop-Tools". As default the installer also wants to install a graphical environment. I deselected this, because it is not necessary to install the graphic packages on a server, where you can't use it. So the installation will finish earlier and the system stays free of unnecessary packages.

2.8 Bootloader Installation

After installing all selected packages, the installer wants to install the bootloader. The installer asks if it should install the bootloader to the master boot record. It is very important to choose NO as answer, because installing the boot loader to your master boot record means, that your master boot record on your hard disk will be overwritten. So you can't boot your system on your hard disk any longer. After selecting No as answer it is possible to specify the path for the bootloader installation. Because the USB stick is mounted on device `/dev/sdb` and `/boot` is on `/dev/sdb1` this is the right path. After this step, the installation is completed, and the system is now ready for operating.

3 Initial boot

The first boot of a new installed system is always an exiting moment. Now it is possible to see, if the setup was really successful and in this case, if the system was really installed on the right device. To boot the USB stick, the boot sequence has to be changed in the BIOS. The first boot device should now be the USB key.

3.1 Problems by booting from the USB stick

I didn't have any problems by booting my Debian from the USB stick. However, there are a lot of possible errors which can occur during the initial boot. The most frequent error is a problem with the boot loader. Because I didn't have problems, I don't want to dwell into these errors. In general, it is possible to solve each problem by using the Internet. There are a lot of different possible solutions, and in general these solutions work well.

3.2 Additional packages

After your system booted successfully for the first time, the Debian system is up and running. So the task for this week is done. Nevertheless, I decided to install some further features to have a more userfriendly system. Therefore, I installed `sudo`, so I can deactivate the root user and I have a safer system. Additionally I installed `vim`, which is my preferred text editor. Sadly the `vim` editor is not installed by default anymore. I also decided to install `lynx`, because I want to have a browser for using the world wide web. As I wrote earlier, I decided not to install a graphical environment and therefore I can't use a browser like `firefox`.

Lynx is a console-based browser, which can also be used without a graphical environment.

3.2.1 Installing sudo

As mentioned before, I decided to install sudo with aptitude with the following command:

```
aptitude install sudo
```

It is also possible to use the apt package management system with following command:

```
apt-get install sudo
```

To configure my sudo I run

```
visudo
```

as root and added the line

```
link ALL=(ALL) ALL
```

to the file (in the section "Allow members of group sudo to execute any command"). To activate my sudo account for my user, I run following command:

```
usermod -a -G sudo link
```

Now it is possible to change between user and root, by just adding sudo in front of the command and enter my user password. As I already mentioned, it makes the system safer, because now it is possible to deny login as root directly on the computer or per ssh.

3.2.2 Installing vim

Next, I decided to install vim, because it is my preferred text editor and is also task of the homework. Therefore I use aptitude and my new installed sudo. So I login as user and write to the command line:

```
sudo aptitude install vim
```

After typing in my user password, the system began to install the vim package.

3.2.3 Installing lynx

The last additional package I installed to the system was the web browser lynx.

```
sudo aptitude install lynx
```

With the new lynx browser, I have the possibility to use the world wide web without a graphical environment. So I can always google if I have problems with the system and read the documentation directly on the server.